

Relatório 1º Projeto ASA 2023/2024

Grupo: TP033

Aluno(s): João Gomes (106204) e Sofia Piteira (106194)

Descrição do Problema e da Solução:

O algoritmo que apresentamos é uma solução para o problema de otimização relacionado com o corte de uma placa de madeira em várias partes, tendo como objetivo maximizar o valor do conjunto de peças resultantes. Esta abordagem utiliza uma matriz bidimensional para representar as diferentes combinações de cortes possíveis. Ao percorrer a matriz, o algoritmo calcula iterativamente os valores máximos acumulados anteriormente, considerando cortes horizontais e verticais, de um lado ao outro da chapa. O resultado final indica o valor máximo que pode ser alcançado ao efetuar cortes de forma estratégica na chapa.

O algoritmo:

O resultado final vai ficar no canto inferior direito da matriz. Para calcular esse resultado, temos de calcular o valor de todas as outras posições da matriz. Para calcular o valor de cada posição desta, temos de iterar sobre todos os valores da linha e da coluna até ao quadrado que estamos a preencher. Em cada iteração, calculamos o espaço que sobra e somamos o valor desse índice ao valor da célula atual, guardando-o numa variável temporária caso este seja maior que o valor anteriormente guardado na mesma. Quando acabamos de iterar, verificamos se tal resultado é maior do que o já existente na célula que estamos a preencher e colocamos o maior valor.

Função recursiva:

$$\text{matrix}[i][j] = \begin{cases} 0, & \text{Se } i=0 \vee j=0 \\ \max(\text{matrix}[i][j], \max_{1 \leq k \leq j} (\text{matrix}[i][j-k] + \text{matrix}[i][k])), \\ \max_{1 \leq k \leq i} (\text{matrix}[i-k][j] + \text{matrix}[k][j])), & \text{Caso Contrário} \end{cases}$$

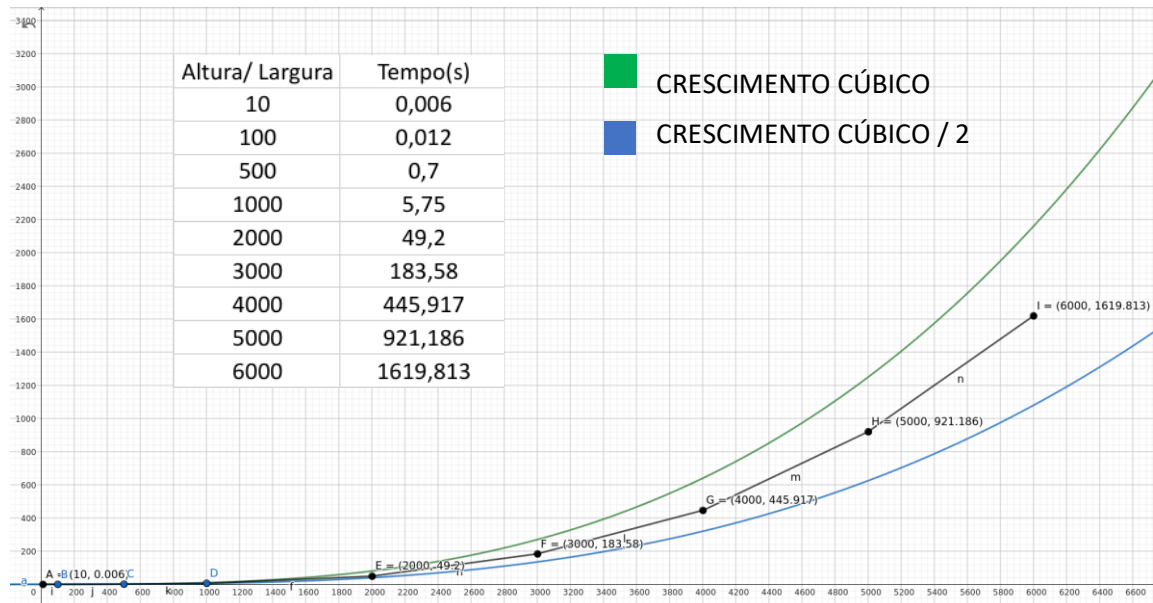
Análise Teórica:

n - Número de coordenadas (X * Y)

- Leitura dos dados de entrada, (X) comprimento, (Y) largura da chapa e número de peças. $O(n)$
- For Loop exterior, na main, que percorre as linhas e posteriormente colunas da matriz bidimensional. $O(n)$
- Primeiro For Loop interior, na main, que calcula o valor para uma posição da tabela iterando sobre todos os valores em X anteriores a essa posição. $O(X^2)$
- Segundo For Loop interior, na main, que calcula o valor para uma posição da tabela iterando sobre todos os valores em Y anteriores a essa posição. $O(Y^2)$

- Apresentação do resultado final. $O(1)$

Segundo a análise teórica, a complexidade total do programa é dominada pelo Loop principal, resultando em $O(X*Y*(X+Y)) = O(2n^3) = O(n^3)$.



O gráfico acima reflete testes do nosso código, o eixo horizontal indica as dimensões iniciais da chapa de madeira e o eixo vertical representa os tempos de execução. Não é relevante para a complexidade total do nosso código, no entanto todos os testes que executamos tinham 200 peças. Com base nestes dados, concluímos que a complexidade do código está entre n^3 e $n^3/2$, o que vai de encontro com a nossa conclusão anterior.



Este gráfico da raiz cúbica do tempo pelas dimensões da tábua vem reforçar a nossa análise teórica, uma vez que é um gráfico praticamente linear.

A complexidade global da solução é $O(n^3)$